(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 676 898 A1**

(12) # EUROPEAN PATENT APPLICATION

(21) Application number: 95104098.9

(51) Int. Cl.⁶: H04N 7/173

(22) Date of filing: 20.03.95

(30) Priority: 04.04.94 US 222781

(43) Date of publication of application: 11.10.95 Bulletin 95/41

(84) Designated Contracting States: AT BE CH DE ES FR GB IT LI NL SE

(71) Applicant: INTERNATIONAL BUSINESS MACHINES CORPORATION
Old Orchard Road
Armonk, N.Y. 10504 (US)

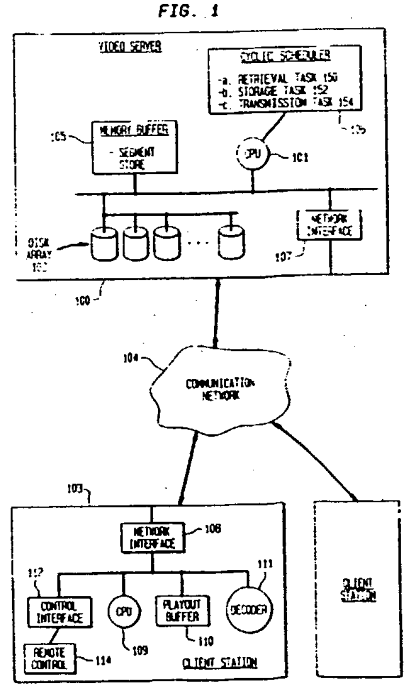
(72) Inventor: Chen, Ming-Syan
710 Brender Lane
Yorktown Heights,
New York 10598 (US)
Inventor: Kandlur, Dilip Dinkar
161 Orchard Road, Apt. 1R
Briarcliff Manor,
New York 10510 (US)
Inventor: Yu, Philip Shi-lung
18 Stornowaye
Chappaqua,
New York 10514 (US)

(74) Representative: Schäfer, Wolfgang, Dipl.-Ing.
IBM Deutschland
Informationssysteme GmbH
Patentwesen und Urheberrecht
D-70548 Stuttgart (DE)

(54) **Frame sampling scheme for video scanning in a video-on-demand system.**

(57) A system and method for performing variable speed scanning or browsing, wherein a user controls the playout speed of a movie, which does not require additional disk or network bandwidth resources. In a preferred embodiment, the method provides for scanning operations for an MPEG video stream. The method satisfies the constraints of the MPEG decoder (in the users set-top box) and require a minimum of additional system resources. The embodiments of the present invention include (a) a storage method, (b1) a segment sampling method, (b2) a segment placement method, and (c) a playout method, where (b1) and (b2) are two alternatives for segment selection. Thus, two sets of solutions are provided to support vriable speed scanning in a disk-array-based video server: One using (a), (b1) and (c), and the other using (a), (b2) and (c).

FIG. 1




EP 0 676 898 A1

## Field of the Invention

The present invention relates to the support of video frame retrieval and scanning in a disk-array-based video server.

## Related Art

In a video on demand (hereinafter, VOD) system, multimedia streams (movies) are stored on a storage server and played out to an end user (receiving) station upon request. The multimedia streams consist of compressed video and audio. The prevalent standard for the video is MPEG (Moving Picture Experts Group). Inter-frame compression techniques such as MPEG provide significant advantages in storage and transmission, and consequently they are universally accepted for VOD applications.

During normal playout, data blocks belonging to the multimedia stream are retrieved from the storage system and transmitted to the receiving station. The receiving station decodes the incoming stream and plays it out. In general, it is desirable to provide the user with VCR--like search functions such as "fast-forward" (also called "scan forward") in a VOD system. There are several conventional approaches to implementing this fast-forward (hereinafter, FF) function, some of which mimic the scan operation of an analog VCR or movie projector. However, each of these approaches imposes additional resource requirements on the system as explained below, where for ease of exposition, it is assumed that the movie has to be scanned at three times of the normal playout rate.

- The multimedia stream is retrieved and transmitted at 3X of the normal playout rate, and the end station filters and plays out the data. This solution requires additional resources (3X the normal rate) at the storage system, the memory buffers, and the network. It also requires additional resources at the end station to process the incoming data.
- The storage system retrieves and transmits every third frame to the end station. This solution requires a significant amount of additional system resources. The multimedia file must now be indexed to retrieve individual frames, and the amount of retrieved data is higher than normal due to the structure of the inter-frame coding.
- The system switches over to a separately coded FF stream to provide the scan operation. This solution, though eliminating any additional read bandwidth or network bandwidth, is extremely expensive in terms of storage cost.

## Summary of The Invention

It is an object of this invention to support fast-forward and fast-backward (scanning) video frame retrieval in a disk-array-based video server.

It is a further object of this invention to support variable speed scanning in video server.

To avoid the previously described drawbacks, the present invention provides a system and method for performing variable speed scanning or browsing, wherein a user controls the playout speed of a movie, which does not require additional disk or network bandwidth resources.

In a preferred embodiment, the method provides for variable speed scanning operations for an MPEG video stream. The method satisfies the constraints of the MPEG decoder (in the users set-top box) and require a minimum of additional system resources. The embodiments of the present invention include (a) a storage method, (b1) a segment sampling method, (b2) a segment placement method, and (c) a playout method, where (b1) and (b2) are two alternatives for segment selection. Thus, two sets of solutions are provided to support variable speed scanning in a disk-array-based video server: One using (a), (b1) and (c), and the other using (a), (b2) and (c).

An MPEG video stream is composed of three types of frames: intra frames (I), predictive frames (P) and interpolated frames (B). The storage method divides a stream of MPEG video frames into media segments. Each segment comprises consecutive frames beginning with an I frame and ending before another I frame. Allocation and retrieval of the multimedia stream is in units of segments. Consecutive segments are stored on different disks in the disk array.

During normal playout, one media segment is retrieved from each disk in a disk array with, say n, disks. These n segments are buffered in memory and transmitted at a fixed rate to the receiving station. In order to preserve smooth operation during the FF mode, n consecutive segments in the scan sequence reside in n different disks.

The segment sampling method, which selectively retrieves segments from a disk array where segments are placed in a round-robin manner, supports segment sampling for FF retrieval, at any FF rate desired by viewers. The segments sampled by this method are uniformly distributed with a minimal variation on the number of segments skipped between every two consecutive retrievals.

The segment placement method is an alternative to the segment sampling method for segment selection. In contrast to the segment sampling method (that is used for a disk-array-based video server where segments are placed in a round-robin manner) the segment placement meth-

od allocates segments to disks judiciously such that no special provision is needed for sampling and the segments can be completely uniformly sampled in an scanning mode for some pre-determined scanning speeds.

The playout method operates the receiving station to play out the retrieved video stream for scanning retrieval. This method, by selectively parsing the incoming media stream and adjusting presentation time-stamps, minimizes the buffer space and the transmission bandwidth required by the server and the end station. These, and other objects, features and advantages of this invention will become apparent from the following detailed description of the invention taken in conjunction with the accompanying drawings.

## Brief Description of the Drawings

Fig. 1 shows a disk-array-based multi-media video server in a network;

Fig. 2 is a block diagram for the procedure of fast-forward retrieval using the segment sampling method;

Fig. 3 is a block diagram for the procedure of fast-forward retrieval using the segment placement method;

Fig. 4 shows a sequence for of MPEG frames;

Fig. 5 shows a disk array where video segments are stored in a round robin manner;

Fig. 6 is a detailed flow diagram for a segment sampling method program;

Fig. 7 shows an example for the segment sampling method;

Fig. 8 is a detailed flow diagram for a segment placement method program;

Fig. 9 shows an example for the segment placement method; and,

Fig. 10 is a block diagram for the playout procedure.

## Detailed Description of A Preferred Embodiment

Fig. 1 shows a disk-array-based video server 100, wherein video data is stored in the disk array 102 and transmitted to the end client stations 103 over the network 104 upon request. The movies (videos) are stored on the disks 102. The video server 100 includes a processor (cpu) 101 which executes tasks under control of a cyclic schedular 106. These tasks include a retrieval task 150 which retrieves videos from the disks 102, a storage task 152 which stores videos temporarily in the memory buffer 105 and a transmission task 154 which transmits videos to client stations 103 by way of a communication network 104 via a network interface 107.

Each of the client stations 103 includes a network interface which provides bi-directional communication between the client station 103 and the video server video the communication network. The client stations also each include processor (cpu) 109 which receives videos via a client station network interface 108 into a playout buffer 110. Each client station 103 also includes a decoder 111 which receives movies from the playout buffer and decodes them for performance on the clients displ. / equipment (such as a television set) and a control interface 112 which receives control commands (including scan speed control commands) from a user via a remote controller 114. These commands are communicated to the video server via the network interface 108 and the network 104.

The video server 100 can be embodied using any processor of sufficient performance for the number of video streams to be supported. For example, a small capacity video server could be embodied using an RISC System/6000 TM system while a larger capacity server could be embodied using an ES/9000 TM system (both available from International Business Machines Corporation of Armonk, New York). The disk array 102 can be of the RAID level 5 type. The communication network 103 can be, for example, a fiber optic network or a conventional bi-directional cable network. The client stations 103 can be embodied as a set-top box. The Remote control 114 and the control logic can be coupled by way of a conventional infra-red interface. The clients send commands to the server 103 by way of the network 104. In accordance with an embodiment of the present invention, the client can control the scan speed of the video (along a continuous spectrum) by depressing particular buttons on the remote controller 114.

For clarity, the example of fast forward operations will be used in this description. It should be understood however that the described principals apply equally to fast backward (FB) operations.

Fig. 2 illustrates one way to achieve the FF retrieval in a disk-array-based video server, using a storage method 10, a segment sampling method 15, and a playout method 20 according to an embodiment of the present invention. The storage method 10 divides a stream of MPEG video frames into media segments. Each segment comprises consecutive frames beginning with an I frame and ending before another I frame. Allocation and retrieval of the multimedia stream is in units of segments. The segment sampling method 15 selectively retrieves segments from a disk array where segments are placed in a round robin manner and performs fast forward (FF) retrieval at any fast forward speed desired by viewers. The playout method 20 causes the end station to play out the

retrieved video stream from fast forward retrieval and maintains the same buffer space and transmission bandwidth requirement as in the normal play operation.

Fig. 3 illustrates another way to achieve the FF retrieval in a disk-array-based video server, using the storage method 10, the segment placement method 35, and the playout method 20 according to another embodiment of the present invention. The storage method 10 and the playout method 20 are the same as used in conjunction with the method of FIG. 2. The segment placement method 35 allocates segments to disks judiciously such that segments can be completely uniformly sampled in an FF mode for some predetermined FF speeds.

The various tasks and programs used to control scanning will now be described by reference to FIGs. 4-10.

Fig. 4 shows a sequence of MPEG video frames 1-8 that is composed of intra frames (I), predictive frames (P), and interpolated frames (B). The storage order of the frames, which differs from the temporal order, reflects the order in which the frames have to be delivered to the decoder. In the MPEG stream P frames depend on I frames, and B frames depend upon both I and P frames. Hence, it is not possible to play out B frames without the corresponding I and P frames. It is therefore not possible to play out every third frame to achieve 3X playout since this subset would include B frames without the corresponding anchor I or P frames.

For the foregoing reasons, the present storage method divides an MPEG video stream into media segments. Each segment includes consecutive frames beginning with an I frame and ending before another I frame. Allocation and retrieval of the multimedia stream is in units of segments. Consecutive segments are stored on different disks in the disk array 102.

The segment sampling method of the retrieval task 150 will now be described. Consider a disk-array-based video server of the type shown in FIG. 1 in which there are n disks. For the maximal throughput during the normal (standard playback speed) operation, video segments S0-S29 are stored in the disk array 102 in a round-robin manner so that n consecutive segments can be retrieved in one round. Formally, when there are n disks, segment **g** is stored in disk **k=f1(g,n)** where **f1(g,n) = g mod n**. Fig. 5 illustrates an example round-robin segment placement when **n=10**.

To achieve the FF feature, one has to appropriately sample some segments only, while skipping the others. The sampling rate depends upon the FF speed desired. If the FF speed is m times the normal speed, we would in average sample one segment out of m segments. For example, if the FF speed m equals 3, segments s0, s3, s6 and s9 will be retrieved from disk 0, disk 3, disk 6 and disk 9, respectively, in Fig. 5, while skipping segments s1, s2, s4, s5, s7 and s8. The complete sampling procedure will be described in more detail later by reference to the segment sampling method.

To provide the best output during FF, the segments sampled should be as uniformly distributed as possible. For example, to double the speed, one can simply select even segments while skipping the odd ones. However, it can be verified that this method, while uniformly sampling the segments, will not obtain the maximal throughput when there are an even number of disks in the array (such as the situation in Fig. 5 where **n=10**). Specifically, where only the even segments are selected, only half of the disks (i.e., disks with even numbers) participate in the segment retrieval whereas the other half disks (i.e., disks with odd numbers) remain idle. Consequently, to develop an FF video segment retrieval scheme in a disk array where segments are stored in a round-robin manner, we not only want to sample the segments as uniformly as possible but also need to ensure that the maximal throughput is attained. Such a procedure is described below.

Fig. 6 is a flow chart of a program in the retrieval task 150 for determining, for each disk, which video segment to retrieve in an FF mode. Suppose the number of disks in the array is **n** and the FF speed desired is **m** times as the normal operation. Let **lcm(n,m)** denote the least common multiple of **m** and **n**. For example **lcm(3,9)=9**. Let **z=nm/lcm(n,m)**, for notational simplicity. To facilitate the description, an FF segment retrieval for **n=9** and **m=3** is given in Fig. 7, where the segments retrieved are marked with *'s.

Referring now to Fig. 6, the round number of segment retrieval, r, equals one in the beginning (step 60). Depending on whether **r** is an odd number or an even number, each disk will use proper formulas to determine the segment to retrieve (step 70). The case that **r** is an odd number corresponds to the first half of the zig-zag curve resulting from the segments retrieved, such as the line formed by segments 0*, 10* and 20* in the first round of Fig. 7. On the other hand, the case that **r** is an even number corresponds to the second half of the zig-zag curve, such as the line formed by segments 29*, 37* and 45* in the second round of Fig. 7. Specifically, if **r** is an odd number, segment **g** will be retrieved from disk **k**, where **g** is uniquely determined by steps 80 and 85. If r is an even number, segment **g** will be retrieved from disk **k**, where **g** is uniquely determined by steps 90 and 95. After segment **g** is retrieved from disk **k** retrieves in step 100, if more video segments are

needed (step 105), go to step 110 and increase the round number **r** by one. Otherwise, this procedure ends in step 120.

Referring again to Fig. 7, the example for **n=9** and **m=3** is provided and those segments retrieved are marked with "s. Instead of retrieving segments whose numbers are multiples of three (which would make disks 1, 2, 4, 5, 7 and 8 idle), the segment sampling scheme described in Fig. 6 shifts the video segments retrieved in some retrievals. For example, segment 10 is fetched instead of segment 9, and segment 20 is fetched instead of segment 19. Such shifting, though making the segments retrieved not perfectly uniformly distributed, ensures that the maximal throughput is achieved. The group of segments retrieved in the first round of retrieval in Fig. 7 consists of segments 0, 10, 20, 3, 13, 23, 6, 16 and 26, which are then displayed in the order of 0, 3, 6, 10#, 13, 16, 20#, 23, 26, where # indicates a shift for retrieved segment numbers. Under this scheme, the segments retrieved in the second round of retrieval in Fig. 7 are segments 45, 37, 29, 48, 40, 32, 51, 43, and 35, which are then displayed in the order of 29, 32, 35, 37#, 40, 43, 45#, 48, 51. It can be seen from Fig. 7 that the segments retrieved form a zig-zag curve. The number of shifts in a zig-zag is **2nm/lcm(n,m) - 2**, which is the minimum among all FF schemes that could achieve the maximal throughput in a disk-array-based video server where segments are stored in a round-robin manner.

The segment placement method will now be described. As shown in Fig. 2 and Fig. 3, the segment placement method is an alternative to the segment sampling method previously described. In contrast to the segment sampling method (which selectively retrieves segments from a disk array in which segments are stored in a round-robin manner) the segment placement method allocates segments to disks judiciously such that no special provision is needed for sampling and the segment can be completely uniformly sampled in an FF mode for some pre-determined FF speeds. Consider an FF operation in which the playout rate is m times as the normal playout rates. In this FF mode, the sequence of retrieved segments from a given starting segment i is **{i, i+m, i+2m, i+3m, ...}**. Since **n** media segments are retrieved in each round, the segments to be retrieved are **{(r-1)nm, (r-1)nm+m, (r-1)nm+2m, ..., (r-1)nm+(n-1)m}**, where **r** is the round number of segment retrieval. It is necessary to ensure that these segments be mapped to different disks so as to have the maximal throughput. The segment placement function **f2(g,n)**, defines a mapping from media segment **g** to a disk **k**, where **k** is a number ranging from 0 to **n-1** in the disk array with **n** disks.

Assuming that **m** is a sub-multiple of the number of disks **n**, the mapping function can be defined as follows:

**f2(g,n) = (g + [g/n]) mod n**. The first term **(g)** represents a regular scattering of the segments on the **n** disks and the second **(g/n)** represents a skew factor. It can be shown that **f2(g,n)** maps the segments **{(r-1)nm, (r-1)nm+m, (r-1)nm+2m, ..., (r-1)nm+(n-1)m}** to different disks for any **r**.

Referring now to Fig. 8 (which shows a flow chart of a segment placement program in the retrieval task 150), the round number of segment retrieval, **r**, equals one in the beginning (step 200). In the **r-th** round, disk **k** will locate its **i-th** disk (i.e., segment number **g**) to retrieve, where the numbers **i** and **g** are uniquely determined by steps 210 and 220, respectively. After segment **g** is retrieved from disk **k** in step 230, if more video segments are needed (step 240), go to step 250 and increase the round number **r** by one. Otherwise, this procedure ends in step 260.

Referring now to Fig. 9, the placement of video segments in a disk array with 6 disks when the required FF speed is three times as the normal operation is given. This layout of media segments across **n** disks ensures smooth retrieval for playout at different FF speeds if the FF speed-up **m** is a sub-multiple of the number of disks **n**. The layout ensures that the media segments to be retrieved in a round all reside in different disks, so that the load imposed on the storage system by the retrieval process in the FF mode is identical to the load under normal operation.

The playout method is now described. During the normal speed playout, video data is retrieved from the disk in rounds. In each round one media segment is retrieved from each disk in the disk array. The data is buffered temporarily in the server and transmitted at a fixed rate to the end station. In the FF mode, the server retrieves segments based on either the procedure in Fig. 2 or the one in Fig. 3. Referring now to Fig. 10, the processing steps of the receiving station that has the responsibility of parsing the incoming stream and creating a valid input stream for the decoder are given. The receiving end station 103 receives the incoming segment stream from the video server 100 in step 300, and discards intermediate frames that do not have the associated anchor frames in step 310. Note that although the media segment begins with an I frame, it contains B frames that have an anchor frame outside the media segment. These B frames are located immediately after the I frame as shown in Fig. 4, and are ignored since they depend on the last P frame in the preceding media segment which is not retrieved. In step 320 the receiving station adjusts the presentation time-stamps embedded in the stream. The presentation time-

stamp determines the time at which a video frame is to be displayed. It has to be adjusted to compensate for the skipped segments and also for the dropped B frames. so as to reflect the correct playout time. Video frames are then played out in step 330.

This playout policy results in a piecewise continuous playout sequence. It allows the viewer to examine scenes in order to quickly locate the scene of interest. Since the segment size is fixed, no additional buffer or transmission bandwidth is required for the stream. The segment maintains the average data rate of the stream, and hence it is acceptable to the end station decoder.

Now that the invention has been described by way of the preferred embodiment, various modifications and improvements will occur to those of skill in the art. Thus, it should be understood that the preferred embodiment has been provided as an example and not as a limitation. The scope of the invention is defined by the appended claims.

## Claims

1. A method of performing any of variable speed scanning operations in a video server wherein at least one data stream is provided from a mass storage media, comprising the steps of:
   receiving, by the video server, a performance request from a viewer for showing a particular video;
   transmitting a first data stream carrying the particular video from the video server to reception equipment at the viewer's location;
   receiving, by the video server, a scanning request including control data indicative of a scanning speed;
   identifying, as a function of the control data, particular frames of the video to be retrieved from the mass storage media;
   transmitting, to the viewers location, a second data stream consisting essentially of only the particular frames, instead of the first data stream.

2. The method of Claim 1 wherein the frames are identified and encoded intra (I) frames, predictive frames and interpolated frames and stored in the mass storage media as units of retrieval each of the units of retrieval comprising consecutive ones of the frames beginning with an I frame and ending before another I frame.

3. The method of Claim 1 wherein mass storage media is a disk array, the particular frames are also identified as a function of a number of disks in the array, the frames of the video are distributed among a plurality of disks in the disk array and wherein a plurality of the frames are retrieved from the disks in parallel.

4. The method of Claim 1 wherein the storage media is an array of disks and comprising the further step of distributing the frames among the disks in the array as a function of a number of disks in the array and at least one predetermined scanning rate.

5. The method of Claim 1 wherein mass storage media is a disk array, and wherein the frames of the video are distributed among a plurality of disks in the disk array and such that the particular frames are distributed evenly among the disks and wherein the total number of frames retrieved is the same as for transmission of the first data stream.

6. The method of Claim 3 wherein the frames are stored as segments on the media and wherein each of the segments comprises a sequence of consecutive frames which can be decoded as a unit.

7. A video-on-demand system, comprising:
   a user station having a video decoder, a control interface for receiving scan speed control commands from a user and a first network interface for receiving video data from the video server and for communicating the scan speed control commands to the video server;
   a video server having a mass storage media, a second network interface for sending the video data to the user station and for receiving the control commands from the user station; and,
   processing logic responsive to the control commands, for identifying particular frames of a video to be retrieved from the mass storage media and provided to the user station; and
   a communication network, coupled to the first network interface and the second network interface.

8. The video-on-demand system of Claim 7 wherein the frames are stored as segments on the media and wherein each of the segments comprises a sequence of consecutive frames which can be decoded as a unit by the user station.

9. A video server, comprising:
   a mass storage media for storing a plurality of videos;
   a network interface for sending video data to a user, for receiving a request for performance of one of the videos and scan speed control commands from the user; and,

processing logic responsive to the control commands, for identifying particular frames of the one of the videos to be retrieved from the mass storage media and provided to the user, wherein particular frames of the video are skipped as a function of speed information embedded in the scan speed control commands.

10. The video server of Claim 9 wherein the frames of the one of the videos are identified and encoded as intra (I) frames, predictive frames and interpolated frames and stored in the mass storage media as units of retrieval each of the units of retrieval comprising consecutive ones of the frames beginning with an I frame and ending before another I frame.

11. The video server of Claim 9 wherein mass storage media is a disk array, the particular frames are also identified as a function of a number of disks in the array, the frames of the video are distributed among a plurality of disks in the disk array and wherein a plurality of the frames are retrieved from the disks in parallel.

12. The video server of Claim 9 wherein the storage media is an array of disks and wherein the frames a distributed among the disks in the array as a function of a number of disks in the array and at least one predetermined scanning speed other than a normal playout speed.

13. The video server of Claim 9 wherein the mass storage media is a disk array, and wherein the frames of the video are distributed evenly among disks in the disk array such that the number of frames retrieved during scanning is the same as for transmission of a video stream at normal playout speed.

14. The video server of Claim 9 wherein the frames are stored as segments on the media and wherein each of the segments comprises a sequence of consecutive frames which can be decoded as a unit by a user station.

15. A video server, comprising:
an array of disks having a plurality of movies stored therein, each movie being stored in the array in units of retrival comprising consecutive intra (I) frames, predictive frames and interpolated frames, each unit of retrieval comprising consecutive ones of the frames beginning with an I frame and ending before another I frame;
a network interface for sending video data to a user, for receiving a request for performance of one of the movies and scan speed control commands from the user; and,
processing means responsive to the control commands, for identifying particular frames of the one of the movies to be retrieved from the mass storage media and provided to the user,
retreival means for retrieving the particular frames of the one of the movies from the disks in parallel groups of the frames, wherein the number of frames retrieved in parallel from the disks during scanning is the same as for transmission of a video stream at normal playout speed.

FIG. 1

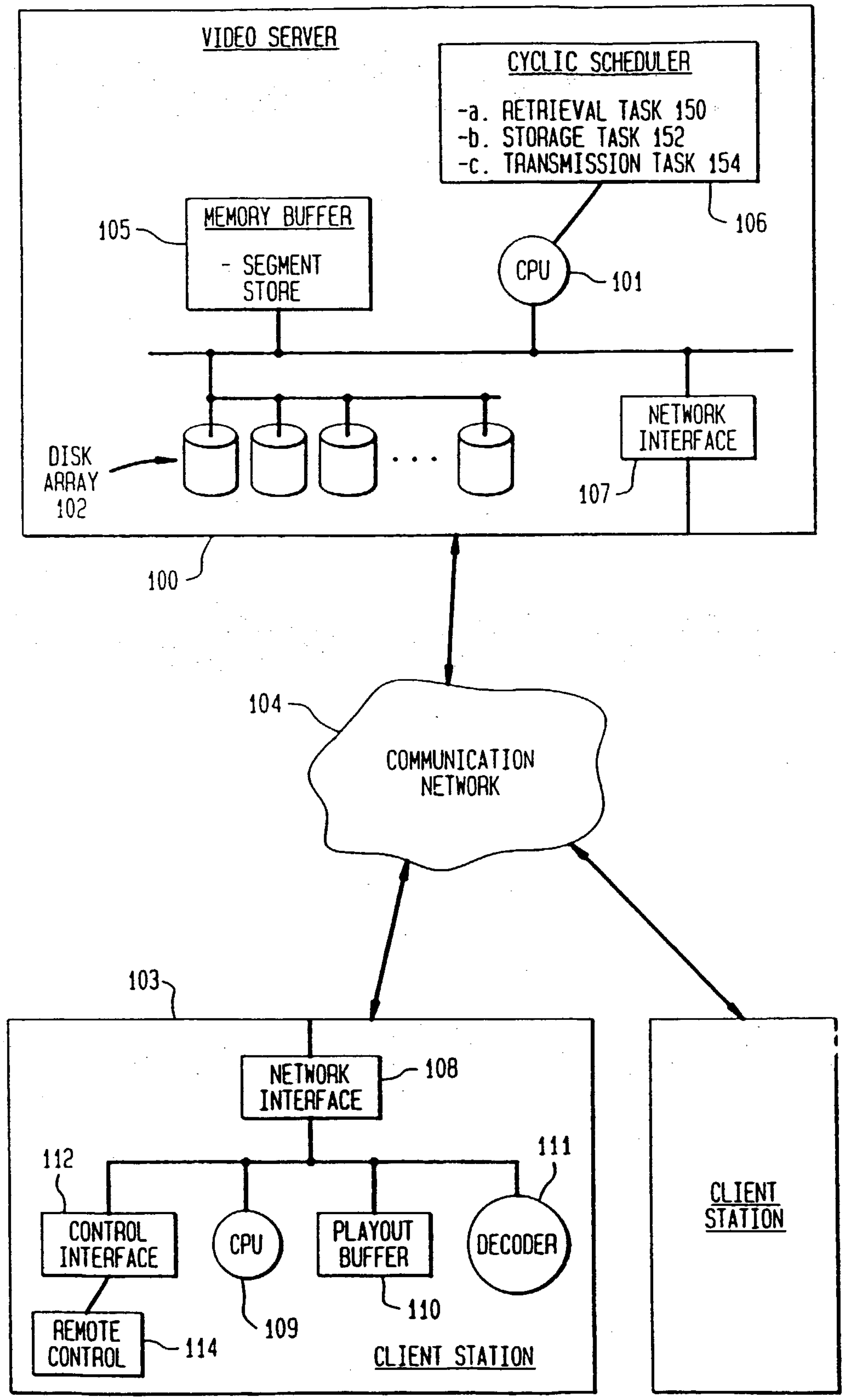

VIDEO SERVER
CYCLIC SCHEDULER
-a. RETRIEVAL TASK 150
-b. STORAGE TASK 152
-c. TRANSMISSION TASK 154
106
MEMORY BUFFER
- SEGMENT STORE
105
CPU
101
DISK ARRAY 102
NETWORK INTERFACE
107
100
104
COMMUNICATION NETWORK
103
NETWORK INTERFACE
108
112
111
CONTROL INTERFACE
CPU
PLAYOUT BUFFER
DECODER
110
109
REMOTE CONTROL
114
CLIENT STATION
CLIENT STATION

FIG. 2

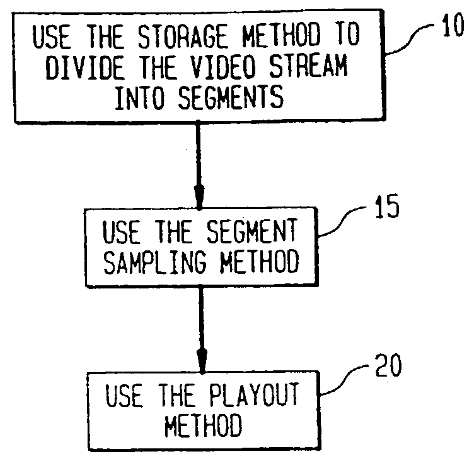

USE THE STORAGE METHOD TO DIVIDE THE VIDEO STREAM INTO SEGMENTS
10
USE THE SEGMENT SAMPLING METHOD
15
USE THE PLAYOUT METHOD
20

FIG. 3

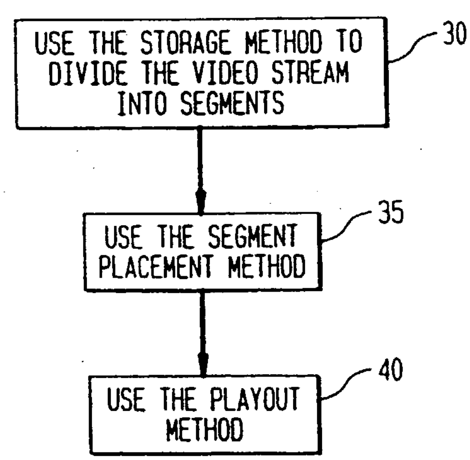
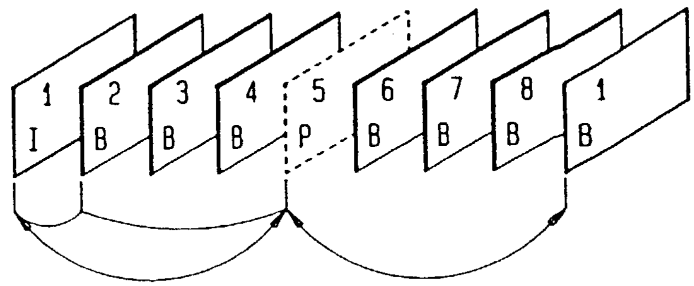

USE THE STORAGE METHOD TO DIVIDE THE VIDEO STREAM INTO SEGMENTS
30
USE THE SEGMENT PLACEMENT METHOD
35
USE THE PLAYOUT METHOD
40

FIG. 4




RELATION BETWEEN I, P AND B FRAMES

TYPICAL FRAME SEQUENCE:
IBBPBBPBBPBBPBBI

STORAGE ORDER
IPBB...

FIG. 5

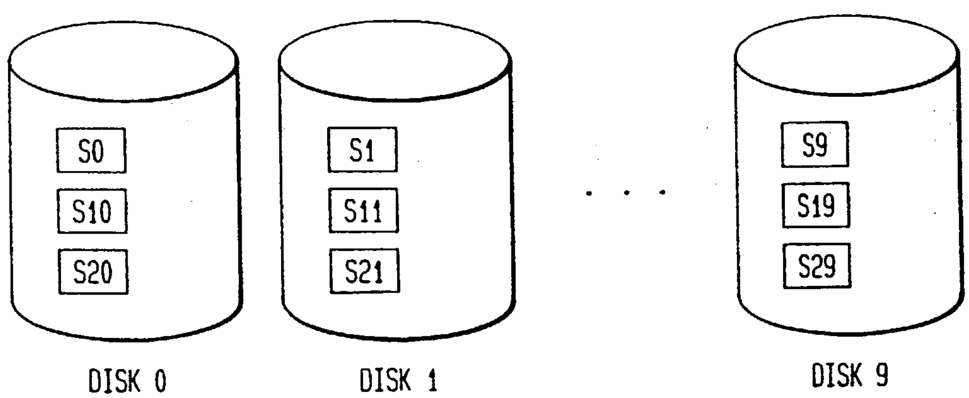


FIG. 6

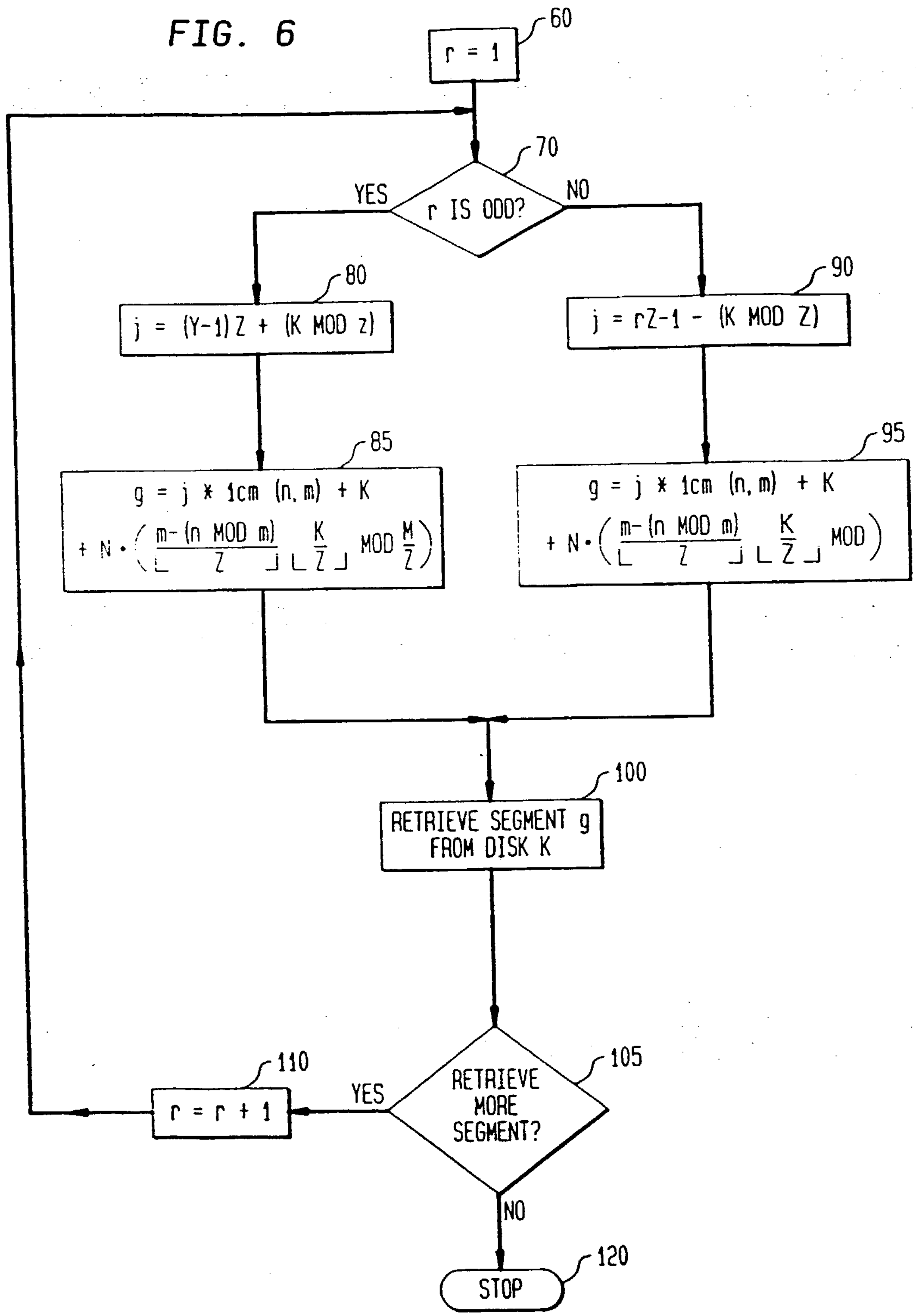

60
r = 1
70
r IS ODD?
YES
NO
80
j = (Y-1) Z + (K MOD z)
90
j = rZ-1 - (K MOD Z)
85
g = j * 1cm (n, m) + K
+ N • ( ⌊(m- (n MOD m))/Z⌋ ⌊K/Z⌋ MOD M/Z )
95
g = j * 1cm (n, m) + K
+ N • ( ⌊(m- (n MOD m))/Z⌋ ⌊K/Z⌋ MOD )
100
RETRIEVE SEGMENT g
FROM DISK K
105
RETRIEVE
MORE
SEGMENT?
YES
110
r = r + 1
NO
120
STOP

FIG. 7

| DISK NO. | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| 1ST ROUND | 0* | 1 | 2 | 3* | 4 | 5 | 6* | 7 | 8 |
| SEGMENTS | 9 | 10* | 11 | 12 | 13* | 14 | 15 | 16* | 17 |
| RETRIEVED (*) | 18 | 19 | 20* | 21 | 22 | 23* | 24 | 25 | 26* |
| 2ND ROUND | 27 | 28 | 29* | 30 | 31 | 32* | 33 | 34 | 35* |
| SEGMENTS | 36 | 37* | 38 | 39 | 40* | 41 | 42 | 43* | 44 |
| RETRIEVED (*) | 45* | 46 | 47 | 48* | 49 | 50 | 51* | 52 | 53 |

A FAST RETRIEVAL FOR n=9 AND m=3

FIG. 8

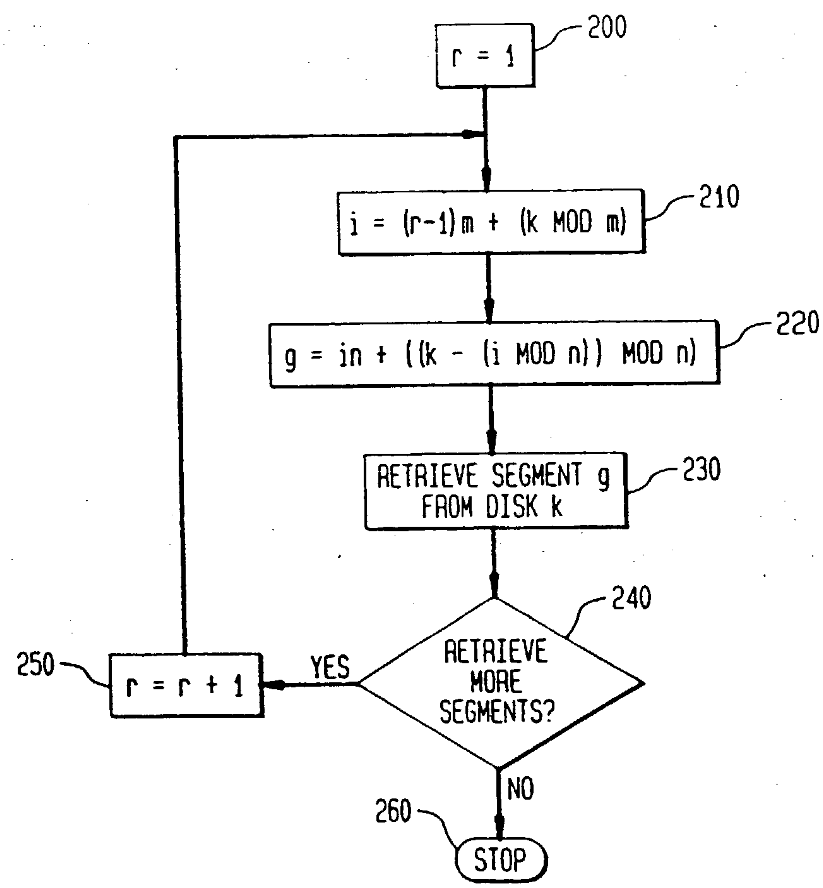


FIG. 9

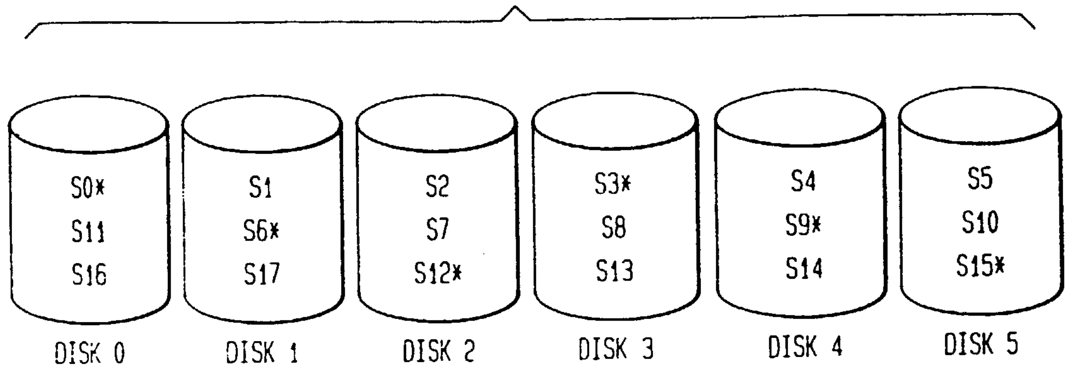
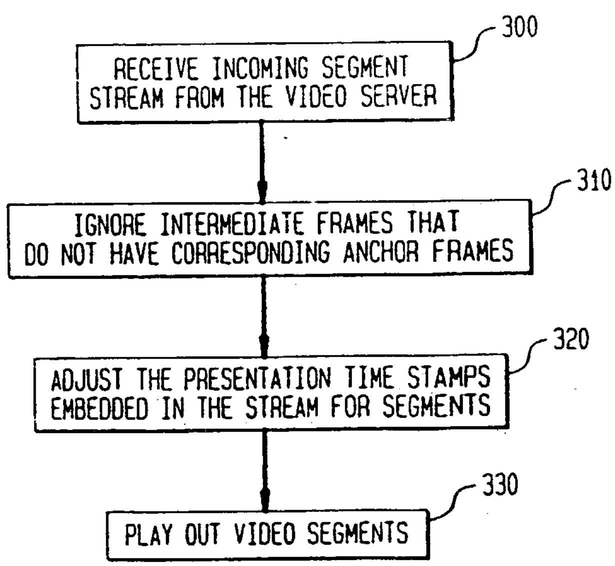

S0*
S11
S16
S1
S6*
S17
S2
S7
S12*
S3*
S8
S13
S4
S9*
S14
S5
S10
S15*
DISK 0
DISK 1
DISK 2
DISK 3
DISK 4
DISK 5

FIG. 10


RECEIVE INCOMING SEGMENT STREAM FROM THE VIDEO SERVER
300
IGNORE INTERMEDIATE FRAMES THAT DO NOT HAVE CORRESPONDING ANCHOR FRAMES
310
ADJUST THE PRESENTATION TIME STAMPS EMBEDDED IN THE STREAM FOR SEGMENTS
320
PLAY OUT VIDEO SEGMENTS
330

European Patent Office

# EUROPEAN SEARCH REPORT

Application Number
EP 95 10 4098

## DOCUMENTS CONSIDERED TO BE RELEVANT

| Category | Citation of document with indication, where appropriate, of relevant passages | Relevant to claim | CLASSIFICATION OF THE APPLICATION (Int.Cl.6) |
|---|---|---|---|
| X | COMMUNICATIONS - RISING TO THE HEIGHTS, DENVER, JUNE 23 - 26, 1991, vol. 2 OF 3, 23 June 1991 INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, pages 842-846, XP 000269608 GELMAN A D ET AL 'A STORE-AND-FORWARD ARCHITECTURE FOR VIDEO-ON-DEMAND SERVICE' * page 843, left column, line 17 - page 846, left column, line 65 * * figures 1-4 * --- | 1-15 | H04N7/173 |
| X<br>Y | US-A-4 949 187 (COHEN) 14 August 1990<br>* column 2, line 18 - line 35 * * column 4, line 1 - column 5, line 17 * * figures 1-4 * --- | 1,7<br>2,9,10, 15 | |
| Y | ELECTRICAL COMMUNICATION, no. 4, 1993 PARIS, FR, pages 402-410, XP 000425706 HOFFMANN ET AL 'Video Compression Techniques for Multimedia Communications' * page 408, left column, line 10 - right column, line 30 * * figure 5 * --- | 2,9,10, 15 | TECHNICAL FIELDS SEARCHED (Int.Cl.6)<br>H04N |
| P,X | US-A-5 371 532 (GELMAN ET AL) 6 December 1994 * abstract * * column 10, line 3 - column 13, line 58 * * figures 2-7 * --- | 1,2,7,9, 10,15 | |
| A | US-A-5 119 188 (MC.CALLEY ET AL) 2 June 1992 * abstract * * column 6, line 50 - column 10, line 68 * * figures 1-12 * ----- | 1-15 | |

The present search report has been drawn up for all claims

| Place of search | Date of completion of the search | Examiner |
|---|---|---|
| THE HAGUE | 12 July 1995 | Van der Zaal, R |

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document